

Testjob templating for LAVA

Milosz Wasilewski, Linaro Ltd.



Outline

- Problem to solve
- LAVA job definition construction
- A bit of history
- Current approach
- Future

Problem to solve

- LKFT test job maintenance
 - 8 device types
 - 20+ 'tests' (test suites or test suite variants)
- Goal: produce valid test job from any device type + test combination

Problem to solve

- Convoluted nature of LAVA job definitions
 - Deployment/device type are connected
 - Sometimes tests depend on deployment type
- Deployments
 - TFTP + NFS
 - U-boot + fastboot
 - Fastboot only
 - U-boot + DFU
 - ...
- Tests
 - Test-definitions
 - Interactive
 - Hidden (boot test)

LAVA job construction

```
timeouts:
  job:
    minutes: 30
  connection:
    minutes: 2
context:
  arch: amd64
  guestfs_interface: virtio
device_type: qemu
job_name: lkft-ltp-timers-master-2194
priority: 25
visibility: public
```



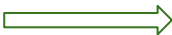
General job settings

```
actions:
- deploy:
  namespace: target
  timeout:
    minutes: 15
  to: tmpfs
  images:
    rootfs:
      image_arg: -drive format=raw,file=(rootfs),if=virtio -m 4096 -smp 4
-nographic
  url: http://example.com/hddimg.xz
  compression: xz
  os: oe
- boot:
  namespace: target
  timeout:
    minutes: 10
  method: qemu
  media: tmpfs
  auto_login:
    login_prompt: 'login:'
    username: root
    login_commands:
      - su
  prompts:
    - 'root@intel-core2-32:'
    - 'root@intel-corei7-64:'
```



Deployment and boot
(device specific)

```
- test:
  namespace: target
  timeout:
    minutes: 20
  definitions:
    - repository: https://github.com/Linaro/test-definitions.git
      from: git
      path: automated/linux/ltp/ltp.yaml
      parameters:
        SKIP_INSTALL: 'true'
        TST_CMDFILES: timers
  name: ltp-timers-tests
```



Test section
(device independent)

General job settings

```
timeouts:  
  job:  
    minutes: 30  
  connection:  
    minutes: 2  
  context:  
    arch: amd64  
    guestfs interface: virtio  
device_type: qemu  
job_name:  
lkft-ltp-timers-master-2194  
priority: 25  
visibility: public
```

- Overall job timeout is declared not calculated
 - Job timeout will terminate the job even if the test timeout is longer
- Context is device specific
 - General job settings depend on device type used

Deployment and boot

- ```
actions:
- deploy:
 namespace: target
 timeout:
 minutes: 15
 to: tmpfs
 images:
 rootfs:
 image_arg: -drive format=raw,file={rootfs},if=virtio -m 4096 -smp 4 -nographic
 url: http://example.com/hddimg.xz
 compression: xz
 os: oe
- boot:
 namespace: target
 timeout:
 minutes: 10
 method: qemu
 media: tmpfs
 auto_login:
 login_prompt: 'login:'
 username: root
 login_commands:
 - su
 prompts:
 - 'root@intel-core2-32:'
 - 'root@intel-corei7-64:'
```
- auto\_login section is build specific
    - Same device can be used to test multiple different builds

# Test section

```
- test:
 namespace: target
 timeout:
 minutes: 20
 definitions:
 - repository: https://github.com/Linaro/test-definitions.git
 from: git
 path: automated/linux/ltp/ltp.yaml
 parameters:
 SKIP_INSTALL: 'true'
 TST_CMDFILES: timers
 name: ltp-timers-tests
```

- Namespaces depend on deployment type
- Timeouts are declared but can vary between devices (some boards are slower)
- Parameters depend on the build/software deployed to the board



# History

- “Static” templates with variable substitution
- Template generation from fragments

# Variable substitution

```
device_type: qemu
job_name: RPB OE boot ${MACHINE} ${MANIFEST_BRANCH} ${BUILD_NUMBER}
timeouts:
 job:
 minutes: 45
 action:
 minutes: 10
 connection:
 minutes: 2
priority: 25
visibility: public
context:
 arch: arm64
 netdevice: tap
 machine: virt
 cpu: cortex-a57
 guestfs_interface: virtio
 extra_kernel_args: "${KERNEL_ARGS}\"
actions:
- deploy:
 namespace: target
 timeout:
 minutes: 15
 to: tmpfs
 images:
 bios:
 image_arg: '-bios {bios}'
 url: https://example.com/QEMU_EFI.fd-AARCH64-RELEASE
 rootfs:
 image_arg: -drive id=disk0,file={rootfs},if=none,format=raw -device
 url: ${EXT4_IMAGE_URL}
 compression: gz
 apply-overlay: true
 sparse: false
 kernel:
 image_arg: -kernel {kernel} --append "console=ttyAMA0 root=/dev/vda rw"
 url: ${KERNEL_URL}
 type: image
 os: oe
```

- Whole template is device and build specific
  - 8\*20 files to maintain
  - Changes specific to device type have to be made in 8 files
- There is no distinction between job sections
- OK for small CI jobs with small number of tests to run
- Hard to maintain for more complex setups (20 different tests executed on several boards)

# Template fragments

```
device_type: gemu
job_name: RPB OE boot ${MACHINE} ${MANIFEST_BRANCH} ${BUILD_NUMBER}
timeouts:
 job:
 minutes: 45
 action:
 minutes: 10
 connection:
 minutes: 2
priority: 25
visibility: public
context:
 arch: arm64
 netdevice: tap
 machine: virt
 cpu: cortex-a57
 guestfs_interface: virtio
 extra_kernel_args: \"${KERNEL_ARGS}\"
actions:
- deploy:
 namespace: target
 timeout:
 minutes: 15
 to: tmpfs
 images:
 bios:
 image_arg: '-bios {bios}'
 url: https://example.com/QEMU_EFI.fd-AARCH64-RELEASE
 rootfs:
 image_arg: -drive id=disk0,file={rootfs},if=none,format=raw -device
 url: ${EXT4_IMAGE_URL}
 compression: gz
 apply-overlay: true
 sparse: false
 kernel:
 image_arg: -kernel {kernel} --append "console=ttyAMA0 root=/dev/vda rw"
 url: ${KERNEL_URL}
 type: image
 os: oe
```

- Job sections are stored in separate files
- Complete job template is composed by joining files
  - Still a lot of files to maintain
  - Not all combinations produce valid LAVA test jobs
- Hard to maintain for more complex setups
- No way to calculate changes between tests like timeouts

# Object based approach

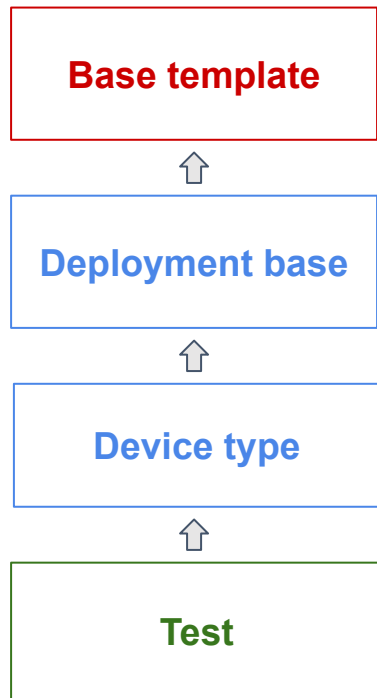
- “Device” object - responsible for deployment part
- “Test” object - responsible for executing the test part
- “Test Plan” - collection of Tests

# Implementation

- Jinja2 based templates
- A set of “Base” templates that allows to minimize amount of code for “Devices”
- Python script that renders jinja2 templates and substitutes variables
  - Direct LAVA submissions
  - Submissions via SQUAD

# Implementation - hierarchy

- Split test job parts between templates
- Render 'test' templates with proper variables set



# Implementation - interface

- Cmdline script
- Parameters:
  - Substitution variables
  - Test plan
  - Device type
- Self tests
  - 'Dry run' option - generates templates but doesn't send them to LAVA or SQUAD
  - LAVA container to validate resulting test jobs

# Implementation - interface

## Example

```
./submit_for_testing.py \
 --variables variables.ini \
 --device-type x15 \
 --dry-run \
 --test-plan lkft-full \
 --test-lava-validity
```

## variables.ini

```
PROJECT_NAME=lkft
KERNEL_BRANCH=master
KERNEL_REPO=https://git.kernel.org/pub/scm/linux/k
ernel/git/next/linux-next.git
KERNEL_COMMIT=139c2d13c258bacc545fc2a4091f7fb0a6fb
08fd
KERNEL_DESCRIBE=next-20191025
MAKE_KERNELVERSION=5.4.0-rc4
```

## Test plan

```
kselftests-native.yaml
kselftests-none.yaml
kselftests.yaml
kvm-unit-tests.yaml
libgpiod.yaml
libhugetlbfs.yaml
ltp-containers.yaml
ltp-cve.yaml
ltp-fs.yaml
ltp-hugetlb-mm.yaml
ltp-io-dio.yaml
ltp-ipc.yaml
ltp-math.yaml
ltp-open-posix.yaml
ltp-sched.yaml
ltp-syscalls.yaml
ltp-timers.yaml
v4l2-compliance.yaml
```



# Base template

```
{% block global_settings %}
{% endblock global_settings %}

device_type: {% block device_type %}{% endblock %}
job_name: {% block job_name %}{% endblock %}
priority: {% block priority %}{% endblock %}
visibility: {% block visibility %}{% endblock %}

metadata:
{% block metadata %}
{% endblock metadata %}

{% block protocols %}
{% endblock protocols %}

actions:
{% block actions %}
{% endblock actions %}
```

- Master template only contains general job section
- There are parts that can be extended further

# Base template - timeouts

```
{# timeouts #}
{% if lxc_deploy_timeout is undefined %}{% set lxc_deploy_timeout = 15 %}{% endif %}
{% if lxc_boot_timeout is undefined %}{% set lxc_boot_timeout = 5 %}{% endif %}
{% if lxc_install_fastboot_timeout is undefined %}{% set lxc_install_fastboot_timeout = 10 %}{% endif %}
{% if target_deploy_timeout is undefined %}{% set target_deploy_timeout = 40 %}{% endif %}
{% if target_boot_timeout is undefined %}{% set target_boot_timeout = 15 %}{% endif %}
{% if test_timeout is undefined %}{% set test_timeout = 60 %}{% endif %}
{% set job_timeout = lxc_deploy_timeout + lxc_boot_timeout + lxc_install_fastboot_timeout +
target_deploy_timeout + target_boot_timeout + test_timeout %}
```

- Timeouts can be calculated for each job separately
- Each part of the job has separate timeout:
  - Deployment
  - Boot
  - Test(s)

# Base template - timeouts

```
{% block global_settings %}
timeouts:
 job:
 minutes: {% if job_timeout is defined %}{{ job_timeout }}{% else %}20{% endif %}
 connection:
 minutes: 2
```

- Job timeout is a sum of all declared timeouts
- There is a hardcoded default in case no other variables are defined
  - The goal is to produce a valid LAVA job

# Base template - deployments

- No code here (too long)
- Offer base templates for common deployment types:
  - Fastboot
  - TFTP + NFS

# Base template - deployments: fastboot

- Fastboot
  - Allow for different partition naming
    - Some boards use 'userdata' while other use 'rootfs'
  - Allow for image manipulation inside LAVA job
    - To add tools/resources required by tests
    - To modify image to fit the board
  - Allow different prompts based on board type or software build
  - Allow to define binary compression type (might differ between CI jobs)

# Base template - deployments: TFTP + NFS

- TFTP+NFS
  - Allow to boot with minimal set (kernel + ramdisk)
  - Allow to add more files to download:
    - dtb
    - u-boot
    - OPTEE
  - Allow for different login options
    - Prompt
    - Password
    - Password change on first login

# Device template

```
{% extends "lkft-fastboot.jinja2" %}

{% set rootfs = true %}
{% set rootfs_label = 'rootfs' %}
{% set DEPLOY_OS = "oe" %}
{% set ROOTFS_URL_COMP = "gz" %}
{% set BOOT_OS_PROMPT = "dragonboard-410c:" %}
{% set auto_login_prompt = "login:" %}
{% set auto_login_username = "linaro" %}

{% block device_type %}dragonboard-410c{% endblock %}

{% block auto_login_commands %}
 login_commands:
 # Become super user to run tests
 - su
{% endblock auto_login_commands %}

{% block deploy_target %}
{{ super() }}
{% include "pre-power-command.jinja2" %}
{% endblock deploy_target %}
```

- Extends base template
- Define device specific variables
- Some variables might be build specific (like compression type)

# Test template - master

```
{% extends device_type %}

{% if PROJECT_NAME is undefined %}{% set PROJECT_NAME = "" %}{% endif %}
{% if BUILD_NUMBER is undefined %}{% set BUILD_NUMBER = "" %}{% endif %}
{% if KERNEL_BRANCH is undefined %}{% set KERNEL_BRANCH = "" %}{% endif %}
{% if OS_INFO is undefined %}{% set OS_INFO = "" %}{% endif %}

{% block job_name %}{{PROJECT_NAME}}-{{OS_INFO}}-{{test_name}}-{{KERNEL_BRANCH}}-{{BUILD_NUMBER}}{% endblock
job_name %}
```

- Defines common variables
- Unifies job naming convention



# Test template

```
{% extends "testcases/master/template-master.jinja2" %}

{% set test_timeout = 25 %}
{% block metadata %}
 {{ super() }}
 libgpiod__url: "{{LIBGPIOD_URL | default('unknown')}}}"
 libgpiod__version: "{{LIBGPIOD_VERSION | default('unknown')}}}"
 libgpiod__revision: "{{LIBGPIOD_REVISION | default('unknown')}}}"
{% endblock metadata %}

{% set test_name = "libgpiod" %}

{% block test_target %}
 {{ super() }}
 - repository: https://github.com/Linaro/test-definitions.git
 from: git
 path: automated/linux/gpiod/gpiod.yaml
 name: libgpiod
{% endblock test_target %}
```

- Each test should work with any device
  - Should at least produce a valid LAVA job
- Tests can be shared between projects

# Additional features

- Test plans for device/project
- Self-test to make sure all test/device combinations render properly
  - This includes checking output job validity with LAVA container
- Variables
  - Can be defined in a file
  - Can be overwritten with command line options
  - Previously variables were inherited from shell environment

# Challenges

- Different requirements from different teams about the same device type (i.e. dragonboard-410c)
  - “Device” part might be team specific
  - Team specific template derives from general-purpose template
- Test and job timeouts
  - LAVA mandates no single timeout is longer than job timeout
  - Timeouts applied to tests might vary between devices
- Prompt matching
  - LAVA matches prompt to detect various events (boot success, test success, etc.)
  - Different builds might use different prompts

# Next

- Add more tests
  - Mainly Android tests (CTS, VTS)
  - Requires enabling Android deployments
- Better self testing
  - There were cases when self tests didn't catch regressions
- Better documentation
  - Documenting variables

# Contributing

- <https://github.com/Linaro/lava-test-plans>
- No separate ML yet but linaro-validation can be used if needed ([linaro-validation@lists.linaro.org](mailto:linaro-validation@lists.linaro.org))
- IRC: #linaro-qa or #lavasoftware on freenode

Thank you!